APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/492,544	01/27/2000	Michael K. Gschwind	Y0999-357(8728-320)	1007

7590   10/01/2003

Frank Chau Esq
F Chau & Associates LLP
1900 Hempstead Turnpike
Suite 501
East Meadow, NY   11554

EXAMINER
MEONSKE, TONIA L

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 10/01/2003

3

Please find below and/or attached an Office communication concerning this application or proceeding.

	Application No.	Applicant(s)
Office Action Summary	09/492,544	GSCHWIND, MICHAEL K.
	Examiner	Art Unit
	Tonia L Meonske	2183

-- The MAILING DATE of this communication appears on the cover sheet with th correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _27 January 2000_ .

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-39_ is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-39_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on _27 January 2000_ is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)☐ The proposed drawing correction filed on _____ is: a)☐ approved b)☐ disapproved by the Examiner.

    If approved, corrected drawings are required in reply to this Office action.

12)☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☐ None of:

        1.☐ Certified copies of the priority documents have been received.

        2.☐ Certified copies of the priority documents have been received in Application No. _____ .

        3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

    a) ☐ The translation of the foreign language provisional application has been received.

15)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☒ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _2_ .

4)☐ Interview Summary (PTO-413) Paper No(s). _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: .

DETAILED ACTION

Specification

1.   The title of the invention is not descriptive. A new title is required that is clearly

indicative of the invention to which the claims are directed.

Drawings

2.   This application lacks formal drawings. The informal drawings filed in this application

are acceptable for examination purposes. When the application is allowed, applicant will be

required to submit new formal drawings.

Claim Objections

3.   Claims 9 and 39 are objected to because of the following informalities: The limitation

"discarding all out-of-order state of a processor the system" appearing in both claims is not

grammatically correct. Appropriate correction is required.

Claim Rejections - 35 USC § 102

4.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the

basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5.   Claims 1, 4, 5, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30,

31, 34, and 35 are rejected under 35 U.S.C. 102(b) as being clearly anticipated by Katzman, US

Patent 3,737,871.

6.   Referring to claim 1, Katzman has taught a method for renaming memory references to

stack locations in a computer processing system, comprising the steps of:

    a.   detecting stack references that use architecturally defined stack access methods (Katzman, Abstract, column 1, lines 20-30, column 4, lines 36-67,); and

    b.   replacing the stack references with references to processor-internal registers (Katzman, column 4, lines 36-67, Column 4, line 35, column 6, line 22,).

7.   Referring to claim 4, Katzman has taught the method according to claim 1, as described above, and further comprising the step of performing a consistency preserving operation for a stack reference that does not use the architecturally defined stack access methods (Katzman, column 4, line 35, column 4, lines 22-47).

8.   Referring to claim 5, Katzman has taught the method according to claim 4, wherein said step of performing a consistency-preserving operation comprises the step of bypassing a value from a given processor-internal register to a load operation that references a stack area and that does not use the architecturally defined stack access methods (Katzman, column 4, line 35, column 4, lines 22-47).

9.   Referring to claim 10, Katzman has taught the method according to claim 1, as described above, and wherein the architecturally defined stack access methods comprise memory accesses that use at least one of a stack pointer, a frame pointer, and an argument pointer (Katzman, column 3, 24-39, column 4, lines 42-48,).

10.   Referring to claim 11, Katzman has taught the method according to claim 1, as described above, and wherein the architecturally defined stack access methods comprise push, pop, and other stack manipulation operations (Katzman, column 4, lines 33-36, column 7, lines 30-41).

11.   Referring to claim 12, Katzman has taught a method for renaming memory references to stack locations in a computer processing system, comprising the steps of

a.   determining whether a load instruction references a location in a local stack using

an architecturally defined register for accessing a stack location (Katzman, column 4,

lines 5-25, Column 4, line 35, column 6, line 22, column 4, line 35, column 6, lines 22-

47);

b.   determining whether a rename register exists for the referenced location in the

local stack, when the load instruction references the location using the architecturally

defined register (Katzman, Abstract, column 1, lines 20-30, column 4, lines 36-67,

column 4, line 35, column 4, lines 22-47); and

c.   replacing the reference to the location by a reference to the rename register, when

the rename register exists (Katzman, column 4, lines 36-67, column 4, line 35, column 4,

lines 22-47).

12.   Referring to claim 13, Katzman has taught the method according to claim 12, as

described above, and wherein the architecturally defined register corresponds to a pointer for

accessing stack locations (Katzman, Abstract, column 1, lines 20-30, column 4, lines 36-67).

13.   Referring to claim 14, Katzman has taught the method according to claim 13, as

described above, and wherein the pointer for accessing the stack locations is one of a stack

pointer, a frame pointer, and an argument pointer (Katzman, column 3, lines 24-39, column 4,

lines 42-48).

14.   Referring to claim 15, Katzman has taught the method according to claim 12, as

described above, and wherein the architecturally defined stack access methods comprise push,

pop, and other stack manipulation operations (Katzman, column 4, lines 33-36, column 7, lines

30-41).

15.   Referring to claim 16, Katzman has taught the method according to claim 12, as described above, and wherein said step of determining whether the renaming register exists comprises the step of computing one of a symbolic address and an actual address of the location (Katzman, Column 3, lines 55-67, Column 4, line 35, column 6, line 22).

16.   Referring to claim 17, Katzman has taught the method according to claim 12, as described above, and wherein said step of determining whether the rename register exists is performed during one of a decode, an address generation, and a memory access phase (Katzman, Column 3, lines 55-67, Column 4, line 35, column 6, line 22).

17.   Referring to claim 18, Katzman has taught the method according to claim 12, as described above, and further comprising the step of performing the load instruction from one of a main memory and a cache of the system, when the rename register does not exist (Katzman, Column 3, lines 55-67, Column 4, line 35, column 6, line 22, column 4, line 35, column 4, lines 22-47).

18.   Referring to claim 19, Katzman has taught the method according to claim 12, as described above, and further comprising the step of determining whether the load instruction references a location in any stack, including the local stack, using another register, when the load instruction does not reference the location using the architecturally defined register (Katzman, column 4, line 35, column 4, lines 22-47).

19.   Referring to claim 21, Katzman has taught the method according to claim 19, as described above, and further comprising the step of performing the load instruction from one of a main memory and a cache of the system, when the load instruction does not reference the location using the other register (Katzman, column 6, lines 47-67).

20.   Referring to claim 22, Katzman has taught the method according to claim 19, as

described above, and further comprising the step of executing a consistency-preserving

mechanism to perform the load instruction from the stack area, when the load instruction

references the location using the other register (Katzman, column 4, line 35, column 4, lines 22-

47).

21.   Referring to claim 23, Katzman has taught the method according to claim 12, as

described above, and further comprising the step of allocating a rename register for the location,

when the rename register does not exist (Katzman, column 4, lines 36-67, Column 4, line 35,

column 6, line 22).

22.   Referring to claim 24, Katzman has taught the method according to claim 23, as

described above, and further comprising the step of inserting an operation, into an instruction

stream containing the load instruction, to load the location from a processor of the system to the

allocated rename register, upon allocating the rename register (Katzman, column 7, lines 1-29,

The operations inserted into the system to perform register renaming.).

23.   Referring to claim 25, Katzman has taught the method according to claim 24, as

described above, and further comprising the step of:

        a.   replacing the reference to the location by a reference to the allocated rename

        register, upon inserting the operation (Katzman, column 7, lines 1-29).

24.   Referring to claim 26, Katzman has taught a method for renaming memory references to

stack locations in a computer processing system, comprising the steps of:

        a.   determining whether a store instruction references a location in a local stack using

        an architecturally defined register for accessing a stack location (Katzman, column 4,

lines 1-25, Abstract, column 1, lines 20-30, column 4, lines 36-67, Column 4, line 35,

column 6, line 22, column 4, line 35, column 4, lines 22-47);

b.   allocating a rename register for the location, when the store instruction references

the location using the architecturally defined register (Katzman, column 4, lines 36-67,

Column 4, line 35, column 6, line 22, column 4, line 35, column 4, lines 22-47); and

c.   replacing the reference to the location by a reference to the rename register

(Katzman, column 4, lines 36-67, Column 4, line 35, column 6, line 22, column 4, line

35, column 4, lines 22-47).

25.   Referring to claim 27, Katzman has taught the method according to claim 26, further

comprising the step of inserting an operation, into an instruction stream containing the store

instruction, to store the location from the rename register to a main memory of the system, upon

replacing the reference to the location by the reference to the rename register (Katzman, column

7, lines 1-29, Column 4, line 35, column 6, line 22, The operations inserted into the system to

perform register renaming.).

26.   Claim 28 does not recite limitations above the claimed invention set forth in claim 19 and

is therefore rejected for the same reasons set forth in the rejection of claim 19 above.

27.   Claim 29 does not recite limitations above the claimed invention set forth in claim 21 and

is therefore rejected for the same reasons set forth in the rejection of claim 21 above.

28.   Claim 30 does not recite limitations above the claimed invention set forth in claim 22 and

is therefore rejected for the same reasons set forth in the rejection of claim 22 above.

29.   Referring to claim 31, Katzman has taught a program storage device readable by

machine, tangibly embodying a program of instructions executable by the machine to perform

method steps for renaming memory references to stack locations in a computer processing

system, the method steps comprising:

  a.   detecting stack references that use architecturally defined stack access methods

  (Katzman, column 4, lines 1-25, Abstract, column 1, lines 20-30, column 4, lines 36-67,

  Column 4, line 35, column 6, line 22,); and

  b.   replacing the stack references with references to processor-internal registers

  (Katzman, column 4, lines 36-67, Column 4, line 35, column 6, line 22,).

30.   Claim 34 does not recite limitations above the claimed invention set forth in claim 4 and

is therefore rejected for the same reasons set forth in the rejection of claim 4 above.

31.   Claim 35 does not recite limitations above the claimed invention set forth in claim 1 and

is therefore rejected for the same reasons set forth in the rejection of claim 1 above.

Claim Rejections - 35 USC § 103

32.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set
> forth in section 102 of this title, if the differences between the subject matter sought to be patented and
> the prior art are such that the subject matter as a whole would have been obvious at the time the
> invention was made to a person having ordinary skill in the art to which said subject matter pertains.
> Patentability shall not be negatived by the manner in which the invention was made.

33.   Claims 2, 3, 32, 33, 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Katzman, US Patent 3,737,871, in view of Morris et al., US Patent 6,286,095.

34.   Referring to claims 2 and 3, Katzman has taught the method according to claim 1, as

described above. Katzman has not specifically taught further comprising the step of

synchronizing an architected state between the processor-internal registers and a main memory

of the computer processing system wherein said synchronizing step comprises the step of

inserting in-order write operations for all of the stack references that are write stack references.

However, Morris et al. have taught synchronizing an architected state between the

processor-internal registers and a main memory of the computer processing system wherein said

synchronizing step comprises the step of inserting in-order write operations for all of the stack

references that are write stack references (Morris et al., Abstract, Figures 4A, 4B, 11, and 12,

column 4, lines 9-37, column 5, lines 14-49) for the purpose of only operating on valid data

(Morris et al., column 6, lines 62-67). It would have been obvious to one of ordinary skill in the

art at the time the invention was made to have the invention of Katzman, include the step of

synchronizing an architected state between the processor-internal registers and a main memory

of the computer processing system wherein said synchronizing step comprises the step of

inserting in-order write operations for all of the stack references that are write stack references,

as taught by Morris et al., for the desirable purpose of only operating on valid data (Morris et al.,

column 6, lines 62-67).

35.   Claim 32 does not recite limitations above the claimed invention set forth in claim 2 and

is therefore rejected for the same reasons set forth in the rejection of claim 2 above.

36.   Claim 33 does not recite limitations above the claimed invention set forth in claim 3 and

is therefore rejected for the same reasons set forth in the rejection of claim 3 above.

37.   Claim 36 does not recite limitations above the claimed invention set forth in claims 2 and

3 and is therefore rejected for the same reasons set forth in the rejection of claims 2 and 3 above.

38.   Claims 6, 8, 9, 20, and 39 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Katzman, US Patent 3,737,871, in view of Wing et al., US Patent 5,926,832.

39.   Referring to claim 6, Katzman has taught the method according to claim 4, as described above. Katzman has not specifically taught and further comprising the step of synchronizing an architected state between the processor-internal registers and a main memory of the computer processing system, and wherein said step of performing a consistency-preserving operation comprises the step of recovering an in-order value for the stack reference from the main memory, upon performing said synchronizing step. However, Wing et al. have taught further comprising the step of synchronizing an architected state between the processor-internal registers and a main memory of the computer processing system (Wing et al., column 26, lines 5-52), and wherein said step of performing a consistency-preserving operation comprises the step of recovering an in-order value for the stack reference from the main memory, upon performing said synchronizing step (Wing et al., column 26, lines 5-52). It would have been obvious to one of ordinary skill in the art at the time the invention was made to have the system of Katzman include the step of synchronizing an architected state between the processor-internal registers and a main memory of the computer processing system, and wherein said step of performing a consistency-preserving operation comprises the step of recovering an in-order value for the stack reference from the main memory, upon performing said synchronizing step, as taught by Wing et al., for the desirable purpose of maintaining consistent data in both memories, so that valid unstale data is used during execution (Wing et al., column 26, lines 5-52).

40.   Referring to claim 8, Katzman, in combination with Wing et al. have taught the method according to claim 6, further comprising the step of writing the in-order value to the main memory in response to a load operation that does not use the architecturally defined stack access methods (Wing et al., column 26, lines 5-52).

41.   Referring to claim 9, Katzman has taught the method according to claim 4, as described

above. Katzman has not taught and wherein said step of performing a consistency-preserving

operation comprises the steps of:

    a.   discarding all out-of-order state of a processor the system;

    b.   synchronizing an architected state between the processor-internal registers and a

main memory of the computer processing system; and

    c.   restarting execution after a store operation has been performed that does not use

the architecturally defined stack access methods.

However, Wing et al have taught:

    a.   discarding all out-of-order state of a processor the system (Wing et al., column 26,

lines 24-30);

    b.   synchronizing an architected state between the processor-internal registers and a

main memory of the computer processing system (Wing et al., column 26, lines 5-52);

and

    d.   restarting execution after a store operation has been performed that does not use

the architecturally defined stack access methods (Wing et al., column 26, lines 24-30).

It would have been obvious to one of ordinary skill in the art at the time the invention

was made to have the system of Katzman, include the steps of discarding all out-of-order

state of a processor the system; synchronizing an architected state between the

processor-internal registers and a main memory of the computer processing system; and

restarting execution after a store operation has been performed that does not use the

architecturally defined stack access methods, as taught by Wing et al., for the desirable

purpose of recovering from memory inconsistencies which occur between memory and

the execution, or rename, registers (Wing et al., column 26, lines 5-52).

42.   Referring to claim 20, Katzman has taught the method according to claim 19. Katzman

has not taught wherein said step of determining whether the load instruction references the

location using the other register comprises the step of marking translation lookaside buffer

(TLB) entries of pages in the local stack as containing stack references, when the load instruction

references the location using the other register. However, Wing et al. have taught wherein said

step of determining whether the load instruction references the location using the other register

comprises the step of marking translation lookaside buffer (TLB) entries of pages in the local

stack as containing stack references, when the load instruction references the location using the

other register (Wing et al., column 22, lines 29-45, column 23, lines 2-45). It would have been

obvious to one of ordinary skill in the art at the time the invention was made to have the

invention of Katzman, utilize the translation lookaside buffer, as taught by Wing et al., as it's an

easy way to keep track of stack references.

43.   Claim 39 does not recite limitations above the claimed invention set forth in claim 9 and

is therefore rejected for the same reasons set forth in the rejection of claim 9 above.

44.   Claims 7, 37, and 38 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Katzman, US Patent 3,737,871, in view of Wing et al., US Patent 5,926,832, and Morris et al.,

US Patent 6,286,095.

45.   Referring to claim 7, Katzman, in combination with Wing et al. have taught the method

according claim 6, as described above. They have not taught wherein the in-order value is

written to the main memory by an in-order write operation inserted into an instruction stream

containing an instruction corresponding to the stack reference, when the stack reference is a

write stack reference. However, Morris et al. have taught the in-order value is written to the

main memory by an in-order write operation inserted into an instruction stream containing an

instruction corresponding to the stack reference, when the stack reference is a write stack

reference (Morris et al., Abstract, Figures 4A, 4B, 11, and12, column 4, lines 9-37, column 5,

lines 14-49) for the purpose of only operating on valid data (Morris et al., column 6, lines 62-67).

It would have been obvious to one of ordinary skill in the art at the time the invention was made

to have the invention of Katzman, include wherein the in-order value is written to the main

memory by an in-order write operation inserted into an instruction stream containing an

instruction corresponding to the stack reference, when the stack reference is a write stack

reference, as taught by Morris et al. for the desirable purpose of only operating on valid data

(Morris et al., column 6, lines 62-67).

46.   Claim 37 does not recite limitations above the claimed invention set forth in claim 7 and

is therefore rejected for the same reasons set forth in the rejection of claim 7 above.

47.   Claim 38 does not recite limitations above the claimed invention set forth in claim 8 and

is therefore rejected for the same reasons set forth in the rejection of claim 8 above.

Conclusion

48.   Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Tonia L Meonske whose telephone number is (703) 305-3993.

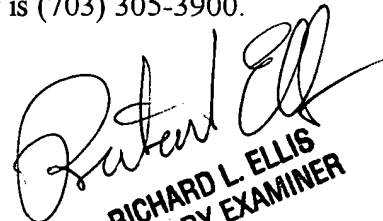The examiner can normally be reached on Monday-Friday, 9-6:30, with every other Friday off.

49.   If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Eddie P Chan can be reached on (703) 305-9712. The fax phone numbers for the

organization where this application or proceeding is assigned are (703) 746-7239 for regular

communications and (703) 746-7238 for After Final communications.

50.   Any inquiry of a general nature or relating to the status of this application or proceeding

should be directed to the receptionist whose telephone number is (703) 305-3900.

tlm
September 25, 2003

RICHARD L. ELLIS
PRIMARY EXAMINER